Knowledge-based Collision Avoidance

F. P. Coenen, G. P. Smeaton and A. G. Bole

(*Liverpool Polytechnic*)

This paper describes preliminary studies towards a navigational Knowledge-based System currently in progress within the School of Engineering and Technology Management at Liverpool Polytechnic.

1. INTRODUCTION. Ever since the advent of computer technology, many attempts have been made to produce a computerized solution to the collision avoidance problem.[1-3] Although a large number of possible approaches have been suggested, to date a satisfactory solution has yet to be achieved. This has been largely due to the algorithmic approach, generally consisting of mathematical modelling of the encounter situation, pursued. This approach worked well for open water two-ship encounter scenarios, but as soon as multi-ship encounters were considered the models became increasingly more complex, and a satisfactory solution was never arrived at. Constraints imposed by the presence of land masses and chart features, ownship passage plan and the likely manoeuvres of targets were therefore not seriously considered.

The principal reason for the failure of the algorithmic methods was that seafarers themselves did not take this approach, and here lies the probable key to the computerized collision avoidance problem. Essentially the seafarer solves collision avoidance problems by referring to a bank of knowledge accumulated over years of training and practical experience. This was appreciated by earlier researchers but the limitations of conventional computing techniques in using information that defied rigorous and precise description rendered the approach of academic value only.

However, over the last decade we have seen the rapid rise of expert systems technology, of which the earliest example is arguably the GPS (General Problem Solver) system which was created by Newall, Shaw and Simon in 1957.[4] Expert systems are designed to emulate human experts in a specific domain using knowledge culled from these experts. Hence they are often referred to as intelligent knowledge-based systems (IKBS). Today there are many expert systems in existence of which the best known is probably the MYCIN system[5] which identifies antibiotic agents to counteract bacterial infections. Two principal expert system languages have also been evolved, namely LISP developed by John McCarthy in the USA in 1956,[6] and Prolog developed by Alan Colmerauer in France in 1972.[7]

The advent of expert systems technology thus prompted the School of Engineering and Technology Management at Liverpool Polytechnic into undertaking a three-year research project funded by the Science and Engineering Research Council (SERC), to produce a prototype real-time navigational

 https://doi.org/10.1017/S0373463300015125

knowledge-based system. The main function of the system is to advise the 'on watch' navigator on all aspects of navigation. The project was divided into several sub-projects including the design of an open water, multi-ship encounter, knowledge-based, collision avoidance system. The design of this system is now nearing completion and thus forms the subject of this paper. A block diagram illustrating the system's architecture is given in Fig. 1.

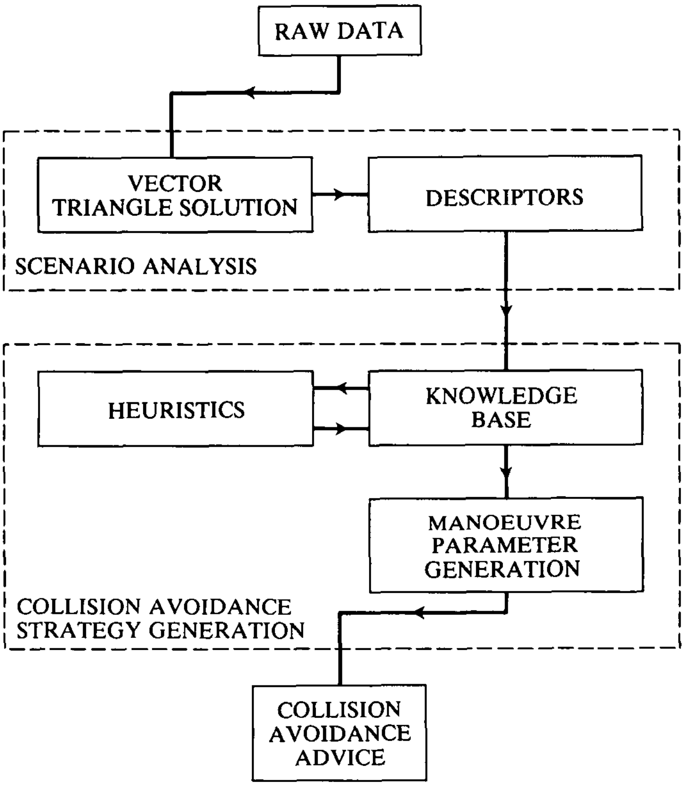

Fig. 1. Schematic for knowledge-based collision avoidance system

2. SYSTEM DESIGN. The foundation of the system is a rule base designed to reflect the accumulated experience and practical know-how of the expert mariner. The rule base was compiled by reference to:

(*a*) The International Regulations for Preventing Collisions at Sea 1972;[8]
(*b*) The interpretation given to these Regulations by mariners, navigational experts and the maritime legal profession; and
(*c*) The practice of good seamanship.

The required information was culled from these sources and translated into a useable form through a process referred to as knowledge engineering. This is the art of extracting information concerning a particular domain from experts and literature and moulding this information into a chosen knowledge formalism. There are many different formalisms for presenting knowledge in existence. The formalism adopted for use with the system was that of if-then production rules.[9] The principal advantage of using this representation is its modularity. Additional rules can be added or existing rules amended simply and efficiently without adversely affecting the rest of the knowledge base.

The knowledge engineering process for the last two of the above knowledge sources consisted of presenting the recognized experts with specific two-ship encounter situations to be solved. Six specific encounter types were identified for this purpose.

(1) Target crossing from port to starboard.
(2) Target crossing from starboard to port.
(3) Target overtaking ownship.
(4) Ownship overtaking Target.
(5) Target on reciprocal course.
(6) Target stopped.

Once an expert had proposed a solution to the basic two-ship encounter, a further target was added to the scenario in such a way that the original target was still the 'most hazardous' target, but so as to preclude the original solution. An alternative solution was thus required. This process was repeated by adding further targets in the same manner until ownship was completely boxed in and thus unable to manoeuvre.

As expected the experts did not always agree. However, on investigation of the results it became apparent that certain set-piece manoeuvres were associated with the six situation types. To date this has resulted in the identification of 14 possible collision avoidance actions, a number of which are associated with each type of situation.

These actions were prioritized for each situation type according to their desirability. The most desirable was given a rule order number of 0, the next most desirable an order number of 1, and so on.

The above knowledge was then represented within the system by a series of production rules of the form:

if A and B then C

Where:

A = The situation type
B = The rule order number
C = The collision avoidance action

An action is thus selected for the most hazardous target according to the situation type and the rule order number. The effect that any additional targets may have on the selection is contained in a set of heuristics of the form:

if A and B then (reject/accept Action)

Where:

A = The proposed action
B = A description of the additional target in question

Heuristics can be thought of as rules of thumb, educated guesses, intuition or common-sense judgements.[10] As such they are not guaranteed to be right but give good results in a large number of cases. These heuristics take into account the following considerations in relation to any additional target.

(1) Location in relation to ownship.
(2) The degree of risk involved, if any.
(3) The nature of the target's movement.
(4) The expectancy, if any, of a target manoeuvre.

Thus the system will search through the data base triggering each rule in turn until the situation type associated with the most hazardous target and the current rule order number are matched successfully, thus causing the rule to be fired. Once a rule is fired the system proceeds as directed by the rule. First the viability of the attached action is tested using the heuristics. If the proposed action is found to be unsuitable the action is rejected. This causes the system to back track, increment the rule order number by 1 and re-enter the rule base. Otherwise the system proceeds to carry out any further processing that may be required to derive the parameters associated with the selected action.

The resulting strategy is then output to the user in the form of collision avoidance advice together with an option to display a detailed explanation of how the strategy was arrived at. If there are no hazardous targets, no collision avoidance advice will be output.

3. RAW DATA. The raw data utilized by the system are based on standard information made available by the solution of target vector triangles. The information thus derived adequately describes the nature of a target's movement. However, the description is made up of some 13 individual pieces of information, including the initially received eastings and northings and the domain associated with the target. Furthermore a practising mariner is accustomed to thinking in terms of phraseology such as 'crossing from starboard to port, passing ahead', rather than maintaining a mental list of numeric factors. Such descriptive phrases are also widely utilized by the International Regulations for Preventing Collisions at Sea. A similar approach was adopted for use with the system and each target is thus allocated a descriptor referred to as its status descriptor. A descriptor can be thought of as a plain language description of information originally defined numerically. The status descriptor has two parts: a primary part and a secondary part. The primary part consists of a general description based on the Collision Avoidance Regulations. The secondary part adds to the first by making it more specific. For example, the primary descriptor may consist of 'crossing from port to starboard' and the second 'passing astern', or in the notation used by the system:

[xing_port_stbd, astern_pass]

There are a total of 10 primary and 10 secondary descriptors although only 32 combinations are valid. The type of situation with which the knowledge base is entered is defined by the primary status descriptor associated with the most hazardous target.

There are also five further descriptors used by the system to give a more complete picture. These are as follows:

(1) The risk status descriptor (R_status) used to describe the degree of risk associated with a target.

(2) The manoeuvre status descriptor (M_status) used to describe whether a target is considered to be manoeuvring or not and if so the nature of the manoeuvre.

(3) The hampered status descriptor (H_status) set according to whether or not the target is considered to be restricted in her ability to manoeuvre within the meaning of the collision avoidance regulations (hampered or unhampered).

(4) The quadrant status descriptor (Q_status) to describe the locality of a target in relation to ownship, for example, close by on the starboard quarter.

(5) The expected target manoeuvre descriptor (ETM status) set according to whether the target is expected to manoeuvre or not and if so the likely nature of this manoeuvre and the reason why.

These descriptors all have default values and, together with the primary status descriptor, are stored in a frame defined for each target. A frame is a data structure which represents an entity type,[11] and consists of a collection of named 'slots' each of which can be filled by values or by a further frame structure. When the slots of a frame are filled the frame is said to be 'instantiated'. A feature of frame data structures is that they allow for the instantiation of default values where no suitable information is available. Thus, for example, in the absence of information to the contrary, targets are always assumed to be unhampered and proceeding on a steady course at a steady speed so as to involve no risk of collision. The use of frames thus allows the system to work with incomplete information.

4. COLLISION AVOIDANCE ACTIONS. During the knowledge engineering process, 14 possible collision avoidance actions were identified. These actions are divided into five action types.

(A) The Stand-on Action:
Stand on (stand_on).

(B) Starboard Alterations:
Alter course to starboard (stbd_alt),
Alter course to starboard and parallel the target (stbd_parallel),
Alter course to starboard so as to place stern on target (stbd_stern_on),
Alter course 40 degrees to starboard immediately (stbd_40), and
Alter course 90 degrees to starboard immediately (stbd_90).

(C) Port Alterations:
Alter course to port (port_alt),
Take a round turn to port (port_round_turn),
Alter course to port so as to place stern on target (port_stern_on),
Alter course 40 degrees to port immediately (port_40), and
Alter course 90 degrees to port immediately (port_90).

(D) Speed reductions:
Speed reduction (reduce_spd), and
Emergency speed reduction (emerg_reduce_spd).

(E) The Default Action:
Emergency (emerg).

The last action type is selected if all else fails, e.g. in a situation where ownship is completely boxed in. The interpretation of this last action is left to the user.

The actions are also divided into standard and emergency actions. Emergency actions are invoked where no suitable standard action can be found or where a target has failed to give way in accordance with the International Regulations for Preventing Collision at Sea. The difference between the two types of strategy is essentially that emergency actions are implemented immediately and require the user/navigator to monitor the situation carefully and continuously. Thus if a last-minute alteration is made by the target in question, a re-appraisal of the situation must be made. The standard actions are as follows:

(1) The ordinary starboard and port course alterations (stbd_alt and port_alt),
(2) The stand-on strategy (stand_on), and
(3) The reduce speed strategy (reduce_spd).

The remaining ten strategies are then considered to be emergency strategies. Emergency strategies are always given rule order numbers of 3 and above (rule order numbers designating and priority associated with an action were described at the end of Section 2). Thus, given an emergency situation, the system can jump straight to the emergency strategies without needlessly first testing the viability of the standard strategies. This approach in itself can also be thought of as a heuristic although an informal one.

Except where stand-on is selected, each action requires some further processing to determine the parameters associated with it. These parameters consist of the magnitude of the proposed action, the time of implementation and the time at which it will be safe for ownship to resume her original course or speed. In some cases the proposed action has a magnitude and/or alteration time already attached or implied; in others all the parameters require calculation.

5. RESTRICTED VISIBILITY. In restricted visibility the special considerations contained in Rule 19 of the International Regulations for Preventing Collisions at Sea are adhered to. This necessitates the use of an alternative restricted visibility rule base and heuristics. The essential difference between the previous 'clear visibility' rule base and the restricted visibility rule base is that the option to stand is not considered in the latter. Similarly, in the case of the restricted visibility heuristics, the restrictions imposed by Rule 19(d) of the Collision Avoidance Regulations are complied with for standard alterations. Essentially this precludes, so far as possible, alterations to port other than for vessels being overtaken and alterations towards vessels abeam or abaft the beam. In the case of emergency alterations, these restrictions will be lifted if no other alternative remains.

6. SYSTEM OPERATION. The operation of the system is illustrated by an example, involving a typical multi-ship encounter, as shown in Fig. 2. The encounter is located in open water and in clear visibility. Target 1 is the most hazardous and thus defines the type of situation, although Target 2 also presents

a threat. A flow chart to accompany the following explanation is given in Fig. 3, and the relevant section of the system's knowledge base for the crossing port to starboard situations type is given in Table 1.

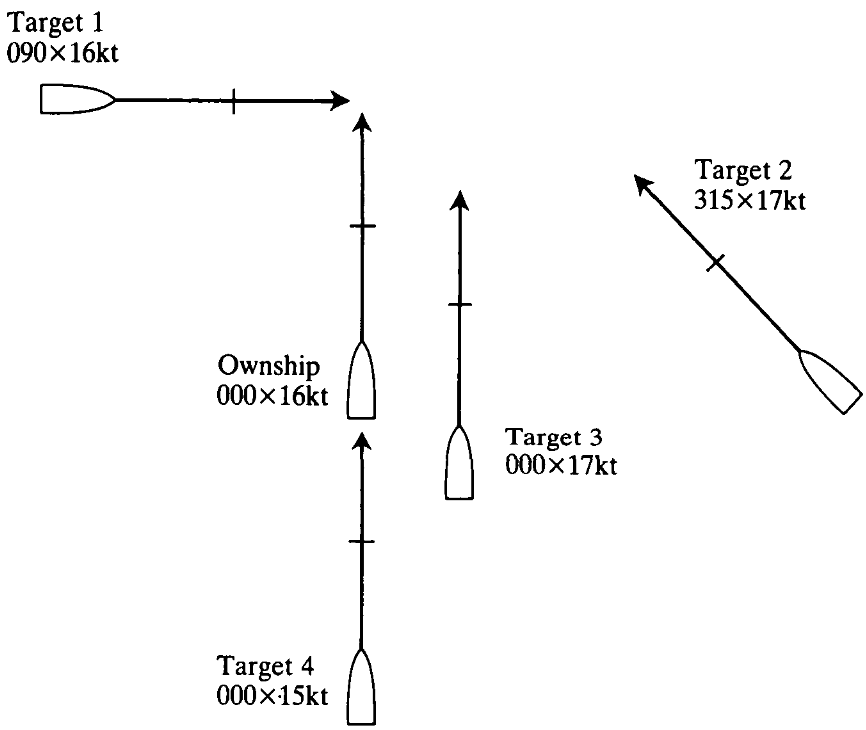

Fig. 2. A typical multi-ship encounter scenario (scale: 1 cm = 1 n.m.)

The rule base is entered with the most hazardous target's descriptors and the rule order number set to 0. A stand-on action is selected first and the efficacy of this action tested heuristically for Targets 2, 3 and 4. The relevant heuristics are given in Table 2. On inspection of these heuristics it can be seen that the proposed initial stand-on action will be rejected because of Target 2.

The rule order number will thus be incremented by 1, the rule base re-entered and the next suitable action selected. In this case this is a standard starboard course alteration. This in turn will be rejected because of the presence of Target 3 which is located close by on ownship's starboard quarter thus precluding all starboard alterations.

In the section of rule base given in Table 1, there is no second-order action. This is because the next most desirable action doubles up as an emergency action and is thus given the rule order number of 3. This is then the point at which the rule base will be entered given an emergency crossing port to starboard situation. The system will thus be unable to 'find' a suitable second-order action. This will have the same effect as if the proposed second-order action was rejected by a

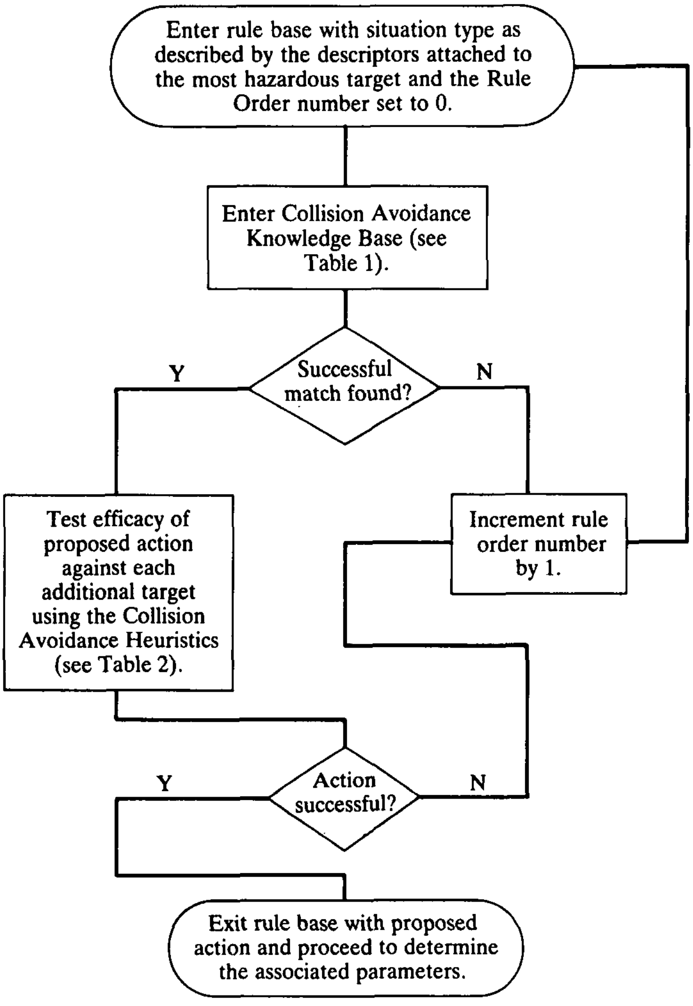

Fig. 3. Flow chart to accompany system operation example

TABLE 1. A SECTION OF THE COLLISION AVOIDANCE KNOWLEDGE BASE

if (order_number = 0) and (Status = [xing_port, stbd,_] and (H_status = [unhampered]) then (stand, on)

if (order_number = 1) and (Status = [xing_port, stbd,_]) and (H_status = [unhampered]) then (stbd, alt)

if (order_number = 3) and (Status = [xing_port, stbd,_]) then (stbd_parallel)

if (order_number = 4) and (Status = [xing_port, stbd,_]) then (reduce_spd)

if (order_number = 5) and (Status = [xing_port, stbd,_]) then (emerg_reduce_spd)

if (order_number = 6) and (Status = [xing_port, stbd,_]) then (port_round_turn),

otherwise (emergency)

heuristic, thus resulting in the system re-entering the rule base with the rule order number incremented by 1.

TABLE 2. A SELECTION OF COLLISION AVOIDANCE HEURISTICS

if (proposed action type = stand_on) and ((Status = [xing_stbd_port,_]) and (M_status = steady) and (R, status = unsafe)) then (reject = action)

if (proposed action type = stbd_alt) and ((Status = [[overtaking_ownship,_]) and (M_status = steady), and (Q_status = close on starboard quarter) then, (reject = action)

if (proposed action type = reduce_spd) and ((M, status = steady) and ETM_status = starboard course, alteration to pass round ownship's stern)) then, (reject = action)

if (proposed action type = reduce_spd) and ((Status, = [overtaken target,_]) and (M_statis = steady) and, (Q_status = located directly astern)) then (reject, = action)

if (proposed action = emerg_reduce_spd) and, ((Status = [overtaken_target,_]) and (M_status = steady) and (Q_status = located directly astern)), then (reject = action)

otherwise (accept action and proceed)

In this case the third-order action is an alteration of course to starboard, to parallel the most hazardous target and observe. This action will then be rejected for the same reason as the standard starboard course alteration was rejected.

The next order action is a standard speed reduction. This will be rejected in turn if an additional target is located directly astern (Target 4), or if it involves ownship reducing speed into the clear water astern into which another target is expected to manoeuvre (Target 1).

The next order action is an emergency speed reduction. This action will only be rejected for targets located directly astern, in this case Target 4.

The only option left, in this example, is a port round turn to pass round the obstructions located astern and on ownship's starboard quarter. The seriousness of the situation overrules the constraints imposed by Rule 17(c) of the International Regulations for Preventing Collisions at Sea. This will then be the action finally proposed, the parameters for which must then be calculated to form the final collision avoidance strategy.

If the port round turn action was also rejected due to another obstruction, for example an additional fifth target proceeding on a reciprocal course close by on ownship's port bow, the only choice left would be the default action. Generally this action will only be called if ownship is completely boxed in.

The collision avoidance strategy generated by the system, if any, is then output to the user as advice. This advice consists of the type of alteration required, the magnitude of the alteration, the time at which it is to be implemented and the time at which it is safe for ownship to resume her original course or speed. The system operates in real time, thus the advice is continuously monitored prior to and during its implementation. If a change in the situation is detected, for example, if a target makes an unexpected course alteration, this will result in a reappraisal of the situation which may in turn cause the advice to be completely or partly updated.

7. ACKNOWLEDGMENTS. The authors are particularly grateful to the Science and Engineering Research Council for their generous financial support for this project. Thanks are also due to Liverpool Polytechnic for the provision of general facilities and support. Many members of the Polytechnic staff and experienced seafarers willingly contributed their expert knowledge; special mention is also made of Dr M. Taylor (University of Liverpool) for his interest in, and help with, the project.

REFERENCES

[1] Kearon, J. (1979). Computer programmes for collision avoidance and track keeping. In *Mathematical Aspects of Marine Traffic* (ed S. H. Hollingdale). Academic Press.

[2] Cannel, W. P. (1981). Collision avoidance as a game of co-ordination. This *Journal*, **34**, 220.

[3] Waddah Abou El-Atta (1984). The assessment and control of risk of collision. Unpublished CNAA Ph.D. thesis, Liverpool Polytechnic.

[4] Sell, P. S. (1985). *Expert Systems, a Practical Introduction*. Macmillan.

[5] Buchanan, B. G. and Shortliffe, E. H. (1984). Rule based expert systems. In *The MYCIN Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley.

[6] Winston, P. H. and Horn, B. K. P. (1981). *LISP*. Addison-Wesley.

[7] Bratko, I. (1986). *Prolog, Programming for Artificial Intelligence*. Addison-Wesley.

[8] I.M.O. (1983). *The International Regulations for Preventing Collisions at Sea*.

[9] Frost, R. A. (1986). *Introduction to Knowledge Based Systems*. Ch. 8, Production Rule Based Systems. Collins.

[10] Pearl, J. (1984). *Heuristics, Intelligent Search Strategies for Computer Problem Solving*. Addison-Wesley.

[11] Minsky, M. (1975). A framework for representing knowledge. In *The Psychology of Computer Vision* (ed. P. H. Winston). Addison-Wesley.